Graph-Based Traffic Analysis for Network Intrusion Detection

Hristo Djidjev, CCS-3; Gary Sandine, T-5

There are two main approaches to detecting malware and intrusion attacks in computer networks: signature-based and anomaly-based. The anomaly detection approach has the advantage that new types of attacks can be identified even before their signatures are discovered and catalogued. Our anomaly-based approach analyzes regular users' activity data from historical NetFlow records and builds a model of normal activity, which we then compare with real-time-activity data. We consider significant deviations from normal historical patterns as anomalies for further investigation as potential cyber attacks.

In this work we focus on modeling the Secure Shell (SSH) traffic in a computer network as a graph, and statistically analyzing various properties of subgraphs corresponding to individual sessions for patterns of normal and anomalous activities.

Our detector operates in two modes. In the off-line (training) mode, it analyzes a database of recorded traffic data and produces a properly organized set of "normal" traffic patterns. In the on-line (detection) mode, it analyzes the between-host traffic in real time, extracting traffic patterns for comparison against historical, normal traffic patterns.

We use NetFlow records from multiple collectors in a large computer network and construct SSH protocol graphs representing SSH traffic observed at LANL in November of 2009. There is a node in the graph for each host (IP address) and a directed edge for each SSH session between the corresponding hosts. Moreover, each edge is labeled with attributes of the session, including session start and end times and data volume. The resulting graph G is a directed multigraph, with multiple edges between the same pair of nodes resulting from different time labels.

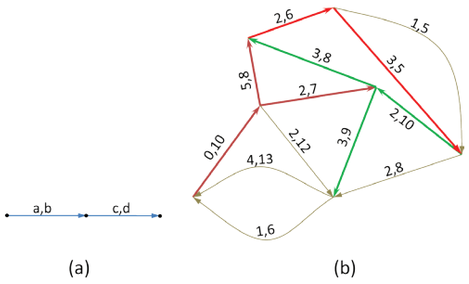

Fig. 1. TSG decomposition. Pairs of numbers on each edge denote the start and end times of the session. (a) Two consecutive edges with time labels a, b and c, d will belong to the same TSG if $a \le c$ and $b \ge d$. (b) Decomposition of a multigraph into TSGs. Edges of the same color belong to the same TSG. Gray thin lines denote single-edge TSGs.

Our objective is to partition G into subgraphs that we call *telescoping graphs* (TSG), which correspond with high probability to a set of interrelated SSH sessions initiated by a single user or attacker (see the example in Fig. 1). Our goal is to represent G as a union of TSGs and to design a very efficient algorithm for computing such a decomposition. We have formally defined the notions of TSG and multigraph decomposition and have shown that such decomposition can be constructed in $O(m \log n)$ time, where m is the number of edges of G and n is the number of nodes. We have implemented our decomposition algorithm and illustrated its efficiency on analyzing traffic logs collected at LANL's internal network. For instance, processing a multigraph of 3.3 million edges takes about 6 seconds on a desktop PC.

Our first approach focuses on the TSG size, depth, and path length distributions. We fit distributions to normal historical data and compare distributions from new sample SSH protocol graphs using the Kullback-Leibler (KL) distance from information theory. KL distance is one way to assign a distance to a pair $(\hat{p}, p)$ of probability mass functions (pmf), an observed pmf $\hat{p}$ corresponding to a sample protocol graph and a historical pmf p. We consider a sample whose pmf has a large KL distance from the historical pmf as a possible anomaly.

We inserted two types of attacks into data collected from the network, first a successful SSH traversal attack resulting in a "caterpillar" subgraph, and second an unsuccessful scan attack resulting in a "spiral out" star-type subgraph. The attack graphs appear in Fig. 2. The attacks were added separately to data for November 5, 2009. The path length distribution analysis (Fig. 3) caused a large KL distance measurement for the sample graph containing the traversal attack, and the size distribution analysis (Fig. 4) resulted in a large KL distance for the sample graph containing the scan attack.

Several authors, (e.g., [1,2]) have used graphs representing network traffic to discover anomalies. In all previous works there have been single graphs (that is, the graph describing all the traffic) to be

For more information contact Gary Sandine at gars@lanl.gov.

analyzed–either locally (e.g., studying the link predicates of a node [2]) or globally (e.g., analyzing the connected component information [1]). In [3] they do consider patterns associated with the nodes that they call graphlets, but they describe the application level activity, rather than intra-host communications. In contrast, we decompose the original protocol graph into subgraphs using time information and analyze the frequency and traffic attributes of these sub-graphs.

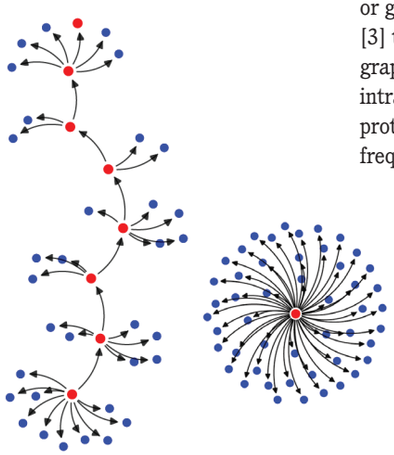

Fig. 2. Traversal and scan attacks.

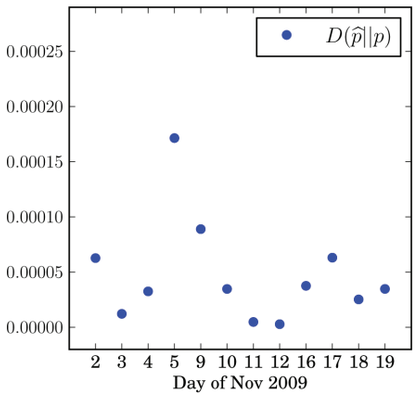

Fig. 3. Traversal attack on 5 Nov.

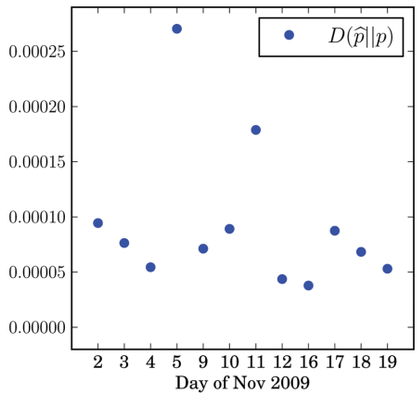

Fig. 4. Scan attack on 5 Nov.

[1] Collins, M., "A Protocol Graph Based Anomaly Detection System," Ph.D. Thesis, Carnegie Mellon University (2008).
[2] Ellis, D., et al., "Graph-based Worm Detection on Operational Enterprise Networks," Technical Report MTR-06W0000035, MITRE (2006).
[3] Karagiannis, T., et al., *SIGCOMM Comput Commun Rev* **35,** 229 (2005).

Funding Acknowledgments
LDRD